



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/580,441

05/22/2006

Peter Neswal

1479-5 PCT/US

2523

23869 7590 03/30/2011  
HOFFMANN & BARON, LLP  
6900 JERICHO TURNPIKE  
SYOSSET, NY 11791

EXAMINER

MITCHELL, JASON D

ART UNIT

PAPER NUMBER

2193

MAIL DATE

DELIVERY MODE

03/30/2011

PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

<b>Office Action Summary</b>	<b>Application No.</b> 10/580,441	<b>Applicant(s)</b> NESWAL, PETER	
	<b>Examiner</b> Jason D. Mitchell	<b>Art Unit</b> 2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 19 January 2011.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-26 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-26 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☒ All    b) ☐ Some \*    c) ☐ None of:
1. ☒ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- |   |   |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)         | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____                                      |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)         | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____   | 6) <input type="checkbox"/> Other: _____                          |

### **DETAILED ACTION**

This action is in response to an amendment filed on 1/19/11.

Claims 1-26 are pending in this application.

### ***Response to Arguments***

#### ***Claim 1:***

In the 1st full par. on pg. 13, the applicant states:

Chefalas discloses a "technique for automatically installing software on one or more network-connected computer systems" in which a server "sends instructions ... to [software agents of the] target computers" at a scheduled time, which "include information such as what software is to be installed, what the hardware and software prerequisites are for that piece of software, and rules for determining how configuration options should be set for the software." (Chefalas Abstract and ¶ 0028). The software agents "install the desired software and any prerequisite software packages onto [the] target computers" and "configure the installed software to operate properly on [the] target computers." (Chefalas Abstract and ¶¶ 0031). However, Chefalas is silent with regard to running a list of rule packages on the client computer a first time to identify installation routines to be run, calling the installation routines, running the list of rule packages on the client computer a second time to identify configuration routines corresponding to the installation routines to be run, and calling their configuration routines, as recited by Claims 1. Therefore, Chefalas fails to disclose the above-identified feature of Claim 1.

The examiner respectfully disagrees. This statement fails to meet the requirements of 37 CFT 1.111(b) in that it does not specifically point out how the language of the claims distinguishes them from the cited reference. Accordingly it is not persuasive.

In the next par. the applicant states:

Applicant notes that the rule packages of the claimed invention are associated with corresponding software components and the rule packages include routines for

Art Unit: 2193

installing the software components and corresponding routines for configuring the software components. By sequentially running the list of rules packages, the rule packages in the list including installation routines are called to run the installation routines and the rules packages in the list including corresponding configuration routines are called to run the configuration routines. Because all the installation routines are initially run in a first and all the configuration routines are run second, correct configuration of all software components is ensured. This overcomes side-effect of overwriting the configuration of underlying software components often encountered by installation operations.

In view of this statement, it is the examiner's understanding that the previous paragraph was intended to assert that Chefalas does not disclose executing each of the rules packages that include installation routines in a first pass, and only after that executing the rules packages that include configuration routines. The examiner respectfully disagrees with such an assertion.

In par. [0031] Chefalas discloses installing all required software ("install the desired software and any prerequisite software packages") and then performing the configuration of these installed programs only once this installation is completed ("then configure the installed software to operate properly"). Accordingly, the claims do not appear to distinguish over the cited reference and the rejection is maintained.

*Claim 7:*

Applicant's arguments on pp. 13-16 with respect to claim 7 have been considered but are moot in view of the new ground(s) of rejection.

In the 1<sup>st</sup> full par. on pg. 16 the applicant states:

Applicant further notes that Applicant has received notable recognition and commercial success for the invention disclosed and claimed in the present

Art Unit: 2193

application. For example, the invention disclosed and claimed in the present application was among the top 5 finalists in the category "IT-Software-Corporate" in The 2010 World Technology Awards held by the World Technology Network (see <http://www.wtn.net/2010winners.html>).

This statement alone is insufficient to establish the level of commercial success or long felt and unresolved need to serve as evidence of non-obviousness.

*Claim 20:*

In the par. bridging pp. 17 and 18 the applicant states:

... Reimert teaches a comparison is made between "previously run installation packages and [a] list of available installation packages", where "[t]hose installation packages listed in both places, duplicated, are removed from the run installation packages list" so that software already installed on a computer is not install on the computer again. (Reimert ¶ 0013). In contrast to Reimert, the claimed invention runs deconfiguration and deinstallation routines for a rules package in response to the rules package being included in lists of successfully run installation and configuration routines, but absent from a framework to be installed and configured. ....

The examiner respectfully disagrees. Here the applicant appears to be arguing primarily that the references do not use the terms "rules packages" and "framework". These terms are only broadly claimed, accordingly it is appropriate to read Chefalas' "instructions 212, 214 and 216" (par. [0028]) on the claimed "rules packages". Further, in view of the breadth of the term "framework" it is reasonable to read Reimert's list of previously run installation packages (e.g. par. [0013] "the previously run installation packages") on the claimed framework. Accordingly in combination these references teach comparing a list of rules packages with a framework to determine what rules form the list of rules packages need to be run (i.e. are not duplicates of those already run).

Art Unit: 2193

*Claim 26:*

Applicant's arguments with respect to claim 26 have been considered but are moot in view of the new ground(s) of rejection.

***Claim Rejections - 35 USC § 112***

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

**Claims 7-12 and 14-23 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.**

**Claim 7** recites in relevant part:

... execution of the instructions by the client computer causes the client computer to perform at least one of:  
a routine for installing this software component on the client computer,  
a routine for deinstalling this software component from the client computer,  
a routine for configuring said software component installed on the client computer, and  
a routine for undoing the configuration of this software component installed on the client computer,  
wherein each routine, *if it establishes an absence requirement* of another software component, branches to a deinstallation routine of another rule package assigned to this other software component.  
(text reformatted for clarity)

From this it should be seen that only one of four alternate routines would "establish[] an absence requirement". Accordingly it is not clear whether or not the "branch[ing] to a deinstallation routine" limitation is required for anticipation.

For the purposes of this examination, in an effort to further prosecution, this limitation will be addressed. However appropriate clarification is required.

**Claims 8-12 and 14-23** depend from claim 7 and are likewise rejected.

**Claim 14** further recites "... the framework including a set of rule packages, a set of detectors for each possible prerequisite ...". This language makes the claim unclear in that it does not indicate what would constitute a "possible" prerequisite. In other words, because an abstract application can depend on any other application, library, object, driver etc. it could be argued that this limitation would require an impossibly large number of detectors.

For the purposes of this examination the claim will be treated as directed to a framework including detectors for each of the software components to be installed as indicated by the framework. However appropriate clarification is required.

**Claims 15-23** depend from claim 14 and are likewise rejected.

**Claim 22** recites in part:

... if [the routine] establishes a presence or absence requirement of another software component, branches to the installation or deinstallation routine of another rule package ...

Parent claim 7 recites in part:

... if [the routine] establishes an absence requirement of another software component, branches to a deinstallation routine of another rule package assigned to this other software component ...

Accordingly claim 22 only requires one or the other of "a presence or absence requirement" wherein parent claim 7 appears to require "an absence requirement".

Accordingly the intended scope of the claims are unclear. Appropriate correction is required.

***Claim Rejections - 35 USC § 102***

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

**Claims 1, 6 and 25 rejected under 35 U.S.C. 102(e) as being anticipated by US 2004/0015961 to Chefalas et al. (Chefalas).**

**Claim 1 and 25:** Chefalas discloses a method for the automatic installation and configuration of software components in a computer network which comprises a plurality of client computers and at least one network resource of installable software components (Abstract “automatically installing software on one or more network-connected computer systems”), comprising the steps of:

a) provision of a framework on the network resource which comprises a rule package for each of the installable software components of the network resource and a list of rule packages to be run, but not the software components themselves (par. [0028] “Server 210 ... sends instructions 212, 214 and 216 to target computers 230, 232, and 234 ... instructions 212, 214 and 216 include information such as ... prerequisites ... configuration options”; par. [0031] “After determinations have been made as to ...



Art Unit: 2193

prerequisites and configuration, agents 220, 222, and 224 download (236, 237, and 238) and install the desired software"),

wherein at least one of the rule packages comprises a routine for loading its software component from the network resource and installing it on a client computer and at least this or one of the other rule packages comprises a routine for configuring its software component installed on a client computer (par. [0031] "agents 220, 222, and 224 download (236, 237, and 238) and install the desired software ... then configure the installed software"),

b) transferring the framework to the client computer (par. [0028] "Server 210 ... sends instructions 212, 214 and 216 to target computers 230, 232, and 234);

c) running the list of rule packages on the client computer a first time to identify installation routines to be run (par. [0031] "agents 220, 222, and 224 download (236, 237, and 238) and install the desired software"; identifying the software is necessary for this step);

d) calling the installation routines (par. [0031] "install the desired software);

e) running the list of rule packages on the client computer a second time to identify configuration routines corresponding to the identified installation routines to be run (par. [0031] "then configure the installed software"; again the claimed identification is necessary, further note that this step occurs after installation); and

f) calling the configuration routines (par. [0031] "then configure the installed software"),

wherein at least step c) is triggered by a local event on the particular client computer (par. [0031] "After determinations have been made as to ... prerequisites and configuration, agents 220, 222, and 224 download (236, 237, and 238) and install the desired software"; par. [0030] "after the information is collected, agents 220, 222, and 224 make a determination as to what version of the software should be installed"; the term 'event' is only broadly claimed, accordingly it is reasonable to interpret it as a specific activity on the client machine; further, in the interest of furthering prosecution, it is noted that those of ordinary skill in the art would understand that communication of the completion of the disclosed collection/determination would obviously be communicated between various objects of an agent using a well known object oriented type event/alert/message (see e.g. the rejection of claim 2)).

**Claim 6:** The rejection of claim 1 is incorporated; further Chefalas discloses at least one of step b) and/or step c) is also triggered by a remote event on the network resource, wherein the remote event includes transmission of a group or broadcast message (par. [0028] "Server 210, at a scheduled time, sends instructions ... to target computers").

### ***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

**Claim 2 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2002/0067504 to Salgado et al. (Salgado).**

**Claim 2:** The rejection of claim 1 is incorporated; further Chefalas does not explicitly disclose step c) is triggered by a system startup or shutdown, system lock or share, user logon or logoff, network logon or logoff, program startup or shutdown, connection or disconnection of hardware or by a timer.

Salgado teaches triggering an installation action with a startup or a timer event (par. [0022] “a program subroutine that will automatically attempt to update the resident printer driver on a workstation or printer server at certain predetermined times or upon the occurrence of certain predetermined events ... (e.g. once a week ... upon startup)”).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to trigger execution of Chefalas’ rule package (par. [0031] “agents 220, 222, and 224 download (236, 237, and 238) and install the desired software”) by a timer (Salgado par. [0022] “a program subroutine that will automatically attempt to update the resident printer driver on a workstation or printer server at certain predetermined times or upon the occurrence of certain predetermined events”). Those of ordinary skill in the art would have been motivated to do so to allow a developer to choose and specify a desirable time to perform the download and update (Salgado par. [0022] “update the

Art Unit: 2193

resident printer driver ... at certain predetermined times”; Chefalas par. [0024] “A convenient time might be one very early in the morning, when most of target computers 130, 132, and 134 will likely not be in use”).

Note that other triggering events would be similarly obvious (see e.g. Salgado par. [0022] “or upon the occurrence of certain predetermined events”)

**Claim 3 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 6,684,242 to Bahlmann (Bahlmann).**

**Claim 3:** The rejection of claim 1 is incorporated; further Chefalas discloses successful installation of a software component on a client computer may have as a prerequisite the presence or absence of another software component or, configuration or deconfiguration of another software component (par. [0028] “instructions 212, 214 and 216 include information such as ... prerequisites ... configuration options”), wherein,

in step a), the framework comprises a detector for each possible prerequisite (par. [0030] “agents 220, 222, and 224 make a determination as to ... what prerequisites are needed”), and,

in step c), if in the course of a rule package it is established by means of a detector that installation or deinstallation, configuration or deconfiguration of another

Art Unit: 2193

software component is necessary, the installation or deinstallation routine, configuration or deconfiguration routine of the rule package assigned to this other software component is called (par. [0044] "the proper version of the software and the necessary prerequisite software components are downloaded").

Chefalas does not explicitly disclose the framework composes at least one rule package which comprises a routine for deinstalling its software component and at least this or one of the other rule packages comprises a routine for undoing the configuration of its software component.

Bahlmann teaches a script comprising a routine for deinstalling and for undoing the configuration of a software component (col. 12, lines 1-8 "the uninstall script 418 restores the configuration settings ... all of the preparation software and information components 400 ... are deleted to complete the uninstall process").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include a rule or rules comprising Bahlmann's deinstallation and configuration scripts (col. 12, lines 1-8 "the uninstall script 418") in Chefalas' framework (par. [0028] "Server 210 ... sends instructions 212, 214 and 216 to target computers 230, 232, and 234). Those of ordinary skill in the art would have been motivated to do so because "[t]here is a possibility that the user 102 will wish to uninstall ... programs

Art Unit: 2193

410 and return the computer 222 to the initial configuration setting." (Bahlmann col. 11, lines 57-59).

**Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2004/0015961 to Bramnick et al (Bramnick).**

**Claim 4:** The rejection of claim 1 is incorporated; further Chefalas does not explicitly disclose detectors for a client computer's hardware or operating system and, in the course of a routine, it is verified by means of such a detector whether the client computer is suitable for the particular installation, deinstallation, configuration or deconfiguration of the software component.

Bramnick teaches verifying by means of a detector for detecting hardware whether a computer is suitable for a particular installation (col. 3, lines 22-26 Automatic application installation procedures ... check for the existence of prerequisite hardware”).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to verify, by means of a detector for detecting hardware, whether a computer is suitable for a particular installation (Bramnick col. 3, lines 22-26 Automatic application installation procedures ... check for the existence of prerequisite hardware”) during the course of a routine (Chefalas par. [0029] "agents 220, 222, and 224 collect information

Art Unit: 2193

concerning which prerequisite software packages must be installed"). Those of ordinary skill in the art would have been motivated to do so in order to ensure that the installed software was able to function correctly (Chefalas par. [0029] "agents 220, 222, and 224 collect information concerning ... prerequisite").

**Claim 5 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0233649 to Reimert (Reimert).**

**Claim 5:** The rejection of claim 1 is incorporated; further Chefalas does not disclose in the course of a routine, it is checked in advance whether the particular installation, deinstallation, configuration or deconfiguration of the software component has already taken place on the client computer and, if so, the routine is immediately terminated.

Reimert teaches checking in advance whether the particular installation of the software component has already taken place on the client computer and, if so, the routine is immediately terminated (par. [0013] "A comparison is made in step 74 between the previously run installation packages and the list of available installation packages found. Those installation packages listed in both places, duplicated, are removed from the run installation packages list").

Art Unit: 2193

It would have been obvious to one of ordinary skill in the art at the time the invention was made to check in advance if a particular installation routine (Chefalas par. [0028] "Server 210 ... sends instructions 212, 214 and 216) has all ready taken place on the client computer (Reimert par. [0013] "Those installation ... duplicated, are removed from the run installation packages list"). Those of ordinary skill in the art would have been motivated to do so to avoid duplicate processing (Reimert par. [0013] "Those installation ... duplicated).

**Claims 7, 11, 13-14, 16-18, 22 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers et al. (Peppers).**

**Claims 7 and 13:** Chefalas discloses a non-transitory machine readable medium storing instructions including a rule package which is executable on an operating system of a client computer for the automatic installation and configuration of software components, which are available on a network resource, on the client computer, wherein the rule package comprises a reference to a software component on the network resource and wherein execution of the instructions by the client computer causes the client computer to perform at least one of: a routine for installing this software component on the client computer (par. [0031] "agents 220, 222, and 224 download (236, 237, and 238) and install the desired software ... then configure the installed software"), a routine for deinstalling this software component from the client



Art Unit: 2193

computer, a routine for configuring said software component installed on the client computer (par. [0028] "Server 210 ... sends instructions 212, 214 and 216 to target computers 230, 232, and 234 ... instructions 212, 214 and 216 include information such as ... configuration options"), and a routine for undoing the configuration of this software component installed on the client computer, wherein each routine.

Chefalas does not disclose a rule package including a routine that establishes an absence requirement of another software component.

Peppers teaches an installation routine that establishes an absence requirement of another software component (par. [0022] "If the process in Step 66 has uncovered one or more software applications that need to be uninstalled") and if the identified software component is detected branching to a deinstallation routine of the software component (par. [0022] "initiate an automatic un-installation of the respective software applications").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include an absence requirement and subsequent deinstallation (Peppers par. [0022] "software applications that need to be uninstalled ... automatic un-installation of the respective software applications") in Chefalas' rule packages (par. [0028] "instructions 212, 214 and 216"). Those of ordinary skill in the art would have been motivated to do so to ensure correct configuration of the target machine (Chefalas

par. [0044] “the proper version of the software and the necessary prerequisite software components are downloaded”; Peppers par. [0013] “Due to potential incompatibilities ... software applications may be required to be ... uninstalled prior to installing the new operating system”).

**Claim 11:** The rejection of claim 7; is incorporated; further Chefalas discloses at least one trigger reference to a remote event on the network resource, wherein execution of the instructions by the client computer causes the client computer to assign at least one of the routines of the rule package to this event based on the at least one trigger reference (par. [0030] “the collected information is forwarded to server 240, and server 240 makes the determinations as to versions, prerequisites, and configuration”; this determination constitutes a remote event; par. [0031] “After determinations have been made ... download and (236, 237, and 238) and install the desired software”).

**Claim 14 and 16-17:** The rejection of claim 7 is incorporated; further Chefalas discloses a framework provided by a network resource in a computer network for execution on a plurality of client computers for the automatic installation and configuration on the client computers of software components available on the network resource (Abstract “automatically installing software on one or more network-connected computer systems”), the framework including a set of rule packages, a set of detectors for each possible prerequisite, and a list of rule packages to be run on the client computers, wherein a prerequisite of successful installation of a software component includes the

Art Unit: 2193

presence or absence of another software component ([0028] "Server 210 ... sends instructions 212, 214 and 216 to target computers 230, 232, and 234 ... instructions 212, 214 and 216 include information such as ... prerequisites"; par. [0030] "make a determination as to ... what prerequisites are needed to be installed").

**Claim 18 and 24:** The rejection of claim 14 is incorporated; further Chefalas discloses a client program which is executed by a client computer to automatically install and configure, on the client computer, software components available on a network resource, wherein execution of the client program runs the list of rule packages to be run in a first pass, calling their installation routines, and runs the list of rule packages to be run in a second pass, calling their configuration routines (par. [0031] "agents 220, 222, and 224 download (236, 237, and 238) and install the desired software ... then configure the installed software").

**Claim 22:** The rejection of claim 18 is incorporated; further Chefalas discloses execution of the client program in conjunction with the rule package causes the client computer to perform at least one of: a routine for installing this software component on the client computer, a routine for deinstalling this software component from the client computer, a routine for configuring said software component installed on the client computer, and a routine for undoing the configuration of this software component installed on the client computer (par. [0028] "instructions 212, 214 and 216 include information such as ... prerequisites ... configuration options"), wherein each routine, if it

Art Unit: 2193

establishes a presence or absence requirement of another software component, branches to the installation or deinstallation routine of another rule package assigned to this other software component, (par. [0044] "the proper version of the software and the necessary prerequisite software components are downloaded") the rule package further containing at least one trigger reference to a remote event on the network resource, wherein the trigger reference assigns at least one of the routines of the rule package to this event, wherein execution of the client program causes the client computer to monitor the occurrence of a remote event on the network resource, including at least one of a transmission of a group and broadcast message, and call the corresponding rule package routine which is assigned via the trigger reference to this event (par. [0030] "server 240 makes the determinations as to versions, prerequisites, and configuration"; this determination constitutes a remote event; par. [0031] "After determinations have been made ... download and (236, 237, and 238) and install the desired software").

**Claims 8 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers et al. (Peppers) in view of US 2004/0015961 to Bramnick et al (Bramnick)**

**Claim 8:** The rejection of claim 7 is incorporated; further Chefalas does not explicitly disclose a reference to a client computer's specific hardware and/or operating system and, and wherein execution of the instructions causes the client computer to verify, by

Art Unit: 2193

means of this reference whether the client computer is suitable for the particular installation, deinstallation, configuration or deconfiguration of the software component.

Bramnick teaches a reference to a computer's specific hardware and causing a client computer to verify, by means of this reference whether a computer is suitable for a particular installation (col. 3, lines 22-26 "Automatic application installation procedures ... check for the existence of prerequisite hardware").

It would have been obvious to one of ordinary skill in the art at the time the invention was made include a reference to a client computer's hardware (Chefalas par. [0029] "information concerning ... prerequisite") and to verify, by means of a detector for detecting hardware, whether a computer is suitable for a particular installation (Bramnick col. 3, lines 22-26 Automatic application installation procedures ... check for the existence of prerequisite hardware") during the course of a routine (Chefalas par. [0029] "agents 220, 222, and 224 collect information concerning which prerequisite software packages must be installed"). Those of ordinary skill in the art would have been motivated to do so in order to ensure that the installed software was able to function correctly (Chefalas par. [0028] "information such as ... what the hardware ... prerequisites are"; par. [0029] "agents 220, 222, and 224 collect information concerning ... prerequisite").

Art Unit: 2193

**Claim 15:** The rejection of claim 14 is incorporated; further Chefalas does not explicitly disclose a framework in conjunction with a rule package comprises a reference to at least one of a client computer's specific hardware and operating system, wherein execution of the instructions by the client computer causes the client computer to verify whether the client computer is suitable for the particular installation, deinstallation, configuration or deconfiguration of the software component by means of the reference, wherein the framework also comprises detectors for a client computer's hardware or operating system and provides the rule packages for the stated verification.

Bramnick teaches a reference to a computer's specific hardware, wherein execution of instructions by the client computer causes the client computer to verify whether a computer is suitable for a particular installation by means of the reference (col. 3, lines 22-26 "Automatic application installation procedures ... check for the existence of prerequisite hardware").

It would have been obvious to one of ordinary skill in the art at the time the invention was made include a reference to a client computer's hardware (Chefalas par. [0029] "information concerning ... prerequisite") and to verify, by means of a detector for detecting hardware, whether a computer is suitable for a particular installation (Bramnick col. 3, lines 22-26 "Automatic application installation procedures ... check for the existence of prerequisite hardware") during the course of a routine (Chefalas par. [0029] "agents 220, 222, and 224 collect information concerning which prerequisite

Art Unit: 2193

software packages must be installed"). Those of ordinary skill in the art would have been motivated to do so in order to ensure that the installed software was able to function correctly (Chefalas par. [0029] "agents 220, 222, and 224 collect information concerning ... prerequisite").

**Claims 9 and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers et al. (Peppers) in view of US 2003/0233649 to Reimert (Reimert).**

**Claim 9:** The rejection of claim 7 is incorporated; further Chefalas does not disclose verifying whether the particular installation, deinstallation, configuration or deconfiguration of the software component on the client computer has already occurred and, if so, terminating its execution.

Reimert teaches verifying in advance whether the particular installation of the software component has already taken place on the client computer and, if so, the routine is immediately terminated (par. [0013] "A comparison is made in step 74 between the previously run installation packages and the list of available installation packages found. Those installation packages listed in both places, duplicated, are removed from the run installation packages list").

Art Unit: 2193

It would have been obvious to one of ordinary skill in the art at the time the invention was made to check in advance if a particular installation routine (Chefalas par. [0028] "Server 210 ... sends instructions 212, 214 and 216) has all ready taken place on the client computer (Reimert par. [0013] "Those installation ... duplicated, are removed from the run installation packages list"). Those of ordinary skill in the art would have been motivated to do so to avoid duplicate processing (Reimert par. [0013] "Those installation ... duplicated).

**Claim 19:** The rejection of claim 18 is incorporated; further Chefalas discloses a rule packages with installation routines and configuration routines.

Chefalas does not disclose a local database which contains a list of rule packages with installation routines which have run successfully and a list of rule packages with configuration routines which have run successfully.

Reimert teaches a local database which contains a list of installation routines which have run successfully (par. [0012] "client computer check in step 56 whether a list of previously run installation packages exists"; the 'checking' action implies database functionality).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to check in advance if a particular installation routine (Chefalas par. [0028]



Art Unit: 2193

“Server 210 ... sends instructions 212, 214 and 216) has all ready taken place on the client computer (Reimert par. [0013] "Those installation ... duplicated, are removed from the run installation packages list"). Those of ordinary skill in the art would have been motivated to do so to avoid duplicate processing (Reimert par. [0013] "Those installation ... duplicated).

**Claims 10 and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers et al. (Peppers) in view of US 2002/0067504 to Salgado et al. (Salgado).**

**Claim 10:** The rejection of claim 7 is incorporated; further Chefalas does not explicitly disclose the rule package contains at least one trigger reference to a local event on the client computer, wherein execution of the instructions causes the client computer to assign at least one of the routines of the rule package to this event based on the at least one trigger reference.

Salgado teaches assigning an installation program containing at least one trigger reference to a local event on a client computer, wherein the trigger reference assigns an installation routine to this event based on the at least one trigger reference (par. [0022] “a program subroutine that will automatically attempt to update the resident printer driver on a workstation or printer server at certain predetermined times or upon the

Art Unit: 2193

occurrence of certain predetermined events ... may be part of a separate upgrade/install program”).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include a reference to a local event (Salgado par. [0022] “a program subroutine that will ... update the resident printer driver ... at certain predetermined times or upon the occurrence of certain predetermined events”) in Chefalas’ rule package (par. [0028] “Server 210 ... sends instructions 212, 214 and 216 to target computers 230, 232, and 234”) to trigger the execution of a routine (par. [0031] “agents 220, 222, and 224 download (236, 237, and 238) and install the desired software).

Those of ordinary skill in the art would have been motivated to do so to allow a developer to choose and specify a desirable time to perform the download and update (Salgado par. [0022] “update the resident printer driver ... at certain predetermined times”; Chefalas par. [0024] “A convenient time might be one very early in the morning, when most of target computers 130, 132, and 134 will likely not be in use”).

**Claim 21:** The rejection of claim 18 is incorporated; further Chefalas discloses a rule package which is executable on an operating system of a client computer for the automatic installation and configuration of software components, which are available on a network resource, on the client computer (Abstract “automatically installing software on one or more network-connected computer systems”), the rule package comprising a reference to a software component on the network resource, wherein execution of the

Art Unit: 2193

client program in conjunction with the rule package causes the operating system of the client computer to perform at least one of: a routine for installing this software component on the client computer, a routine for deinstalling this software component from the client computer, a routine for configuring said software component installed on the client computer, and a routine for undoing the configuration of this software component installed on the client computer (par. [0028] "instructions 212, 214 and 216 include information such as ... prerequisites ... configuration options"), wherein each routine, if it establishes a presence or absence requirement of another software component, branches to the installation or deinstallation routine of another rule package assigned to this other software component (par. [0044] "the proper version of the software and the necessary prerequisite software components are downloaded").

Chefalas does not explicitly disclose the rule package contains at least one trigger reference to a local event on the client computer, wherein the trigger reference assigns at least one of the routines of the rule package to this event wherein the program monitors the occurrence of a local event on the client computer, the local event including at least one of a system startup or shutdown, system lock or share, user logon or logoff, network logon or logoff, program startup or shutdown, connection or disconnection of hardware and response of a timer, and calls the corresponding rule package routine.

Art Unit: 2193

Salgado teaches assigning an installation program containing at least one trigger reference to a local event on a client computer, wherein the trigger reference assigns an installation routine to this event and in response to the event on the client computer calls the corresponding installation program (par. [0022] “a program subroutine that will automatically attempt to update the resident printer driver on a workstation or printer server at certain predetermined times or upon the occurrence of certain predetermined events”).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include a reference to a local event (Salgado par. [0022] “a program subroutine that will ... update the resident printer driver ... at certain predetermined times or upon the occurrence of certain predetermined events”) in Chefalas’ rule package (par. [0028] “Server 210 ... sends instructions 212, 214 and 216 to target computers 230, 232, and 234”) to trigger the execution of a routine (par. [0031] “agents 220, 222, and 224 download (236, 237, and 238) and install the desired software).

Those of ordinary skill in the art would have been motivated to do so to allow a developer to choose and specify a desirable time to perform the download and update (Salgado par. [0022] “update the resident printer driver ... at certain predetermined times”; Chefalas par. [0024] “A convenient time might be one very early in the morning, when most of target computers 130, 132, and 134 will likely not be in use”).

**Claim 12 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers et al. (Peppers) in view of US 6,389,589 to Mishra et al. (Mishra).**

**Claim 12:** The rejection of claim 7 is incorporated; further Chefalas does not disclose the rule package is put in an inactive state in which only deinstallation and deconfiguration routines of the rule package are callable.

Mishra teaches placing a rule package in an inactive state in which only deinstallation and deconfiguration routines are callable (col. 10, TABLE 6 "ACTFLG\_Orphan This application is orphaned. It is no longer deployed, and all existing installs may be left as is"). Those of ordinary skill in the art would have been motivated to do so as a means of indicating and enforcing a determined policy on the client machine (see e.g. col. 10, lines 63-66 "Policy Removal Action Flags, which denote whether the application is set to be Orphaned or Uninstalled when a policy to which the application belongs is removed").

**Claim 20 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers**

**et al. (Peppers) in view of US 2003/0233649 to Reimert (Reimert) in view of US 6,684,242 to Bahlmann (Bahlmann).**

**Claim 20:** The rejection of claim 19 is incorporated; further Reimert teaches the client computer comparing the rule packages entered in the lists with the rule packages contained in the framework (par. [0013] "A comparison is made in step 74 between the previously run installation packages and the list of available installation packages found. Those installation packages listed in both places, duplicated, are removed from the run installation packages list").

Chefalas discloses running an installation routine in a first pass and running a configuration routine in a second pass (par. [0031] "agents 220, 222, and 224 install the desired software ... then configure the installed software"), However, Chefalas and Reimert do not teach running deconfiguration routines in a first pass and deinstallation routines in a second pass.

Bahlmann teaches a script comprising a routine for deinstalling and for undoing the configuration of a software component (col. 12, lines 1-8 "the uninstall script 418 restores the configuration settings ... all of the preparation software and information components 400 ... are deleted to complete the uninstall process").

Art Unit: 2193

It would have been obvious to one of ordinary skill in the art at the time the invention was made to reverse Chefalas' installation and configuration passes (par. [0031] "agents 220, 222, and 224 install the desired software ... then configure the installed software") to undo the installation/configuration using Bahlmann's deinstallation and configuration scripts (col. 12, lines 1-8 "the uninstall script 418"). Those of ordinary skill in the art would have been motivated to do so because "[t]here is a possibility that the user 102 will wish to uninstall ... programs 410 and return the computer 222 to the initial configuration setting." (Bahlmann col. 11, lines 57-59).

**Claim 23 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers et al. (Peppers) in view of Applicant Acknowledged Prior Art Techniques.**

**Claim 23:** The rejection of claim 18 is incorporated; further Chefalas does not disclose a transaction system for each system-modifying component, wherein each system-modifying component includes at least one of the rule packages.

The applicant acknowledges that such transaction systems were known in the art and that their use and benefit would have been clear to those of ordinary skill in the art (pg. 8, lines 27-33 "As a result [of including a transaction system], the system can be rolled back at any time, if for example an installation or configuration fails, as is known in the art").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include a prior art transaction system (AAPA pg. 8, lines 27-33) in Chefalas installation software (e.g. par. [0031] "agents 220, 222, and 224"). Those of ordinary skill in the art would have been motivated to do so simply to achieve the known benefits of such a system (i.e. roll-backs) when an installation causes Chefalas' client computer to begin functioning improperly. In other words, the claimed transaction system does not significantly change the functionality of the installation system nor does the functionality of the transaction system need to be significantly modified in order to interact with the installation system. Accordingly, the inclusion of such a transaction system in this claim merely represents the combination of two prior art system where each provides only the functionality known to be associated with the systems.

**Claim 26 is rejected under 35 U.S.C. 103(a) as being unpatentable over US 2004/0015961 to Chefalas et al. (Chefalas) in view of US 2003/0167354 to Peppers et al. (Peppers) in view of US 6,684,242 to Bahlmann (Bahlmann).**

**Claim 26:** The rejection of claim 1 is incorporated; further Chefalas does not explicitly disclose a prerequisite of successful installation of one of the software components is deconfiguration of another software component that is already installed on the client computer.



Art Unit: 2193

Peppers teaches a prerequisite of successful installation of one of the software components is uninstallation of another software component that is already installed on the client computer (par. [0013] “Due to potential incompatibilities, one or more of the ... software applications may be required to be ... uninstalled prior to installing the new operating system”).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include an absence requirement and subsequent deinstallation (Peppers par. [0022] “software applications that need to be uninstalled ... automatic un-installation of the respective software applications”) in Chefalas' rule packages (par. [0028] “instructions 212, 214 and 216”). Those of ordinary skill in the art would have been motivated to do so to ensure correct configuration of the target machine (Chefalas par. [0044] “the proper version of the software and the necessary prerequisite software components are downloaded”; Peppers par. [0013] “Due to potential incompatibilities ... software applications may be required to be ... uninstalled prior to installing the new operating system”).

Chefalas and Peppers do not explicitly teach a prerequisite of successful installation of one of the software components in deconfiguration of another software component.

Bahlmann teaches uninstalling a software component includes deconfiguration of the software component (col. 12, lines 1-8 “the uninstall script 418 restores the

Art Unit: 2193

configuration settings ... all of the preparation software and information components 400 ... are deleted to complete the uninstall process").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include deconfiguration of a software component (Peppers par. [0022] "software applications that need to be uninstalled ... automatic un-installation of the respective software applications"; Bahlmann col. 12, lines 1-8 "the uninstall script 418 restores the configuration settings") in Chefalas' rule packages (par. [0028] "instructions 212, 214 and 216"). Those of ordinary skill in the art would have been motivated to do so to ensure correct configuration of the target machine (Chefalas par. [0044] "the proper version of the software and the necessary prerequisite software components are downloaded"; Peppers par. [0013] "Due to potential incompatibilities ... software applications may be required to be ... uninstalled prior to installing the new operating system").

### ***Conclusion***

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason D. Mitchell whose telephone number is (571)272-3728. The examiner can normally be reached on Monday-Thursday and alternate Fridays 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bullock Lewis can be reached on (571) 272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Application/Control Number: 10/580,441  
Art Unit: 2193

Page 35

/Jason D. Mitchell/  
Primary Examiner, Art Unit 2193